

Absolvovanie individuálnej odbornej praxe

Individual Professional Practice in the Company

Zadání bakalářské práce

Student:

Tomáš Dižo

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Absolvování individuální odborné praxe
Individual Professional Practice in the Company

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: ABB s.r.o.
2. Struktura závěrečné zprávy:
 - a) Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta.
 - b) Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti.
 - c) Zvolený postup řešení zadaných úkolů.
 - d) Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe.
 - e) Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.
 - f) Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vede odbornou praxi studenta.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Peter Chovanec**

Konzultant bakalářské práce: **Ing. Matěj Liszka**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014



doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Suhlasím so zverejnením tejto bakalárskej práce podľa požiadavkou čl. 26, odst. 9 *Studijného a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

V Ostravě 28. Apríla 2014


.....

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

V Ostravě 28. Apríla 2014


.....

Rád by som poďakoval firme ABB za možnosť vykonávať odbornú prax práve u nich. Chcel by som hlavne poďakovať svojmu konzultantovi Matějovi Liszkovi, môjmu kolegovi Bronislawovi Byrtusovi a taktiež celému kolektívu softwarového oddelenia v ABB. Taktiež by som chcel poďakovať svojmu vedúcemu bakalárskej práce Ing. Petrovi Chovancovi za rady pri písaní tejto bakalárskej práci.

Nakoniec by som chcel poďakovať svojej rodine a priateľom, ktorí pri mne stáli počas celého štúdia.

Abstrakt

Táto bakalárska práca obsahuje popis mojej odbornej praxe vo firme ABB. Práca je rozdelená na 3 časti, prvá popisuje spoločnosť ABB a moje odborné zaradenie v nej. V druhej časti sú opísané použité technológie v projektoch a vykonávané úlohy a ich riešenie. Posledná časť je záver, ktorý obsahuje popis, zhrnutie nových poznatkov a skúseností nadobudnutých počas praxe a celkové jej zhodnotenie.

Kľúčová slova: ABB,prax,ASP.Net,JAVA

Abstract

This work provides a description of my professional experience in company ABB. The work is divided into 3 parts, the first describes ABB and my professional inclusion therein. The second part describes the technologies used in projects and tasks performed and their solutions. The last section contains a summary description of new knowledge and experience gained in practice and its overall assessment.

Keywords: ABB,practice,ASP.Net,JAVA

Seznam použitých zkratk a symbolů

GWT	– Google Web Toolkit
HTML	– Hyper Text Markup Language
CSS	– Cascading Style Sheets
C#	– C-sharp
XML	– eXtensible Markup Language
UI	– User Interface
SQL	– Structured Query Language

Obsah

1	Úvod	5
2	O Spoločnosti ABB	6
2.1	Oblasti podnikania	6
2.2	Pracovné zaradenie študenta	7
3	Použité technológie	8
4	Zadané úlohy ich riešenie	9
4.1	GWT to HTML Tool	9
4.2	Find unused compiled CSS	10
4.3	Analýza projektu ABB MACHSense	10
4.4	RefaktORIZÁCIA	10
4.5	Translators	11
4.6	DataAccess Objekty	11
4.7	Dodatočná refaktORIZÁCIA a urýchlenie aplikácie	12
4.8	AddData Collector	12
4.9	Configure Parameters Css štýly	12
4.10	Testovanie aplikácie	13
4.11	MS UNIT Testy	13
4.12	Coded UI Testy	13
4.13	Technická dokumnetácia	13

5	Záver	15
5.1	Chýbajúce vedomosti a zručnosti	15
5.2	Zhrnutie bakalárskej praxe	15
6	Literatúra	16

Zoznam obrázkov

1	ABB Logo	6
---	--------------------	---

Seznam výpisů zdrojového kódu

1	Príklad definovania CSS štýlov v CSSResources	10
2	Získanie zákazníkov z DB pomocou PetaPoco	11

1 Úvod

Popri veľkom počte zaujímavých tém bakalárskych prác som sa nakoniec rozhodol pre bakalársku prax pretože som si chcel vyskúšať vlastne ako vývoj software prebieha vo veľkých firmách. Taktiež som chcel získať potrebnú prax a získať nové skúsenosti popri štúdiu. Z veľkého zoznamu firiem, ktorý naša škola ponúka som si nakoniec vybral medzinárodnú firmu ABB. Po niekoľkých dňoch od ústneho pohovoru, ktorý pozostával aj z rozhovoru v anglickom jazykom som bol prijatý.

V prvej časti predstavím v krátkosti firmu ABB, jej oblasti podnikania a moje pracovné zaradenie v nej.

V druhej časti popisujem moje zadané úlohy a ich riešenia. Túto časť rozdeľujem na dve menšie podkapitoly prvú kde som pracoval v prvom tíme s mojim poradcom Matějom Liszkom s technológiou JAVA GWT. A druhú kde som pracoval na projekte ABB MACH-Sense v tíme s Bronislawom Byrtusom.

V závere poukazujem na znalosti, ktoré mi chýbali počas riešenia úloh a ktoré som bol nútený doštudovať kvôli úspešnému zvládnutiu zadanej úlohy a celkové zhrnutie mojej bakalárskej práce.

2 O Spoločnosti ABB

ABB je vedúcou spoločnosťou v oblasti technológií prenosu a rozvodu elektrickej energie a automatizačných technológií, ktoré umožňujú zákazníkom zvyšovať výkonnosť a znižovať dopady na životné prostredie. Spoločnosť vznikla v roku 1988 spojením korporácie Allmänna Svenska Elektriska Aktiebolaget (ASEA) a Švajčiarskej spoločnosti Brown, Boveri & Cie (BBC). V súčasnosti skupina ABB pôsobí vo viac ako 100 krajinách a zamestnáva približne 150 000 pracovníkov. V Českej republike pôsobí ABB prostredníctvom svojich výrobkov už od roku 1970 a zamestnáva tu približne 3300 pracovníkov.



Obr. 1: ABB Logo

2.1 Oblasti podnikania

- Výrobky pre energetiku (Power Products) predstavujú kľúčové komponenty pre prenos a rozvod elektrickej energie
- Systémy pre energetiku (Power Systems) poskytujú dodávky systémov a služieb na kľúč pre prenosové a distribučné siete a elektrárne
- Automatizácia výroby a pohony (Discrete Automation and Motion) poskytujú výrobky, riešenie a s nimi súvisiace služby, ktoré zvyšujú priemyslovú produktivitu a energickú účinnosť
- Výrobky nízkeho napätia (Low Voltage Products) vyrábajú a predávajú ističe, spínače, výrobky pre riadiace aplikácie atď., ktoré zaisťujú bezpečnosť osôb a ochranu inštalácií a elektrických prístrojov pred preťažením
- Procesná automatizácia (Process Automation) hlavným cieľom je poskytnúť zákazníkovi výrobky a riešenie pre inštrumentáciu, automatizáciu a optimalizáciu priemyslových procesov

2.2 Pracovné zaradenie študenta

Pri prijímacom pohovore som sa dozvedel, že v Ostravskom Operačnom Centre ABB (CZOPC) pri vývoji množstva projektov využívajú najmä technológiu .NET a jeden pracovník JAVA GWT(v súčasnosti dvaja). Na základe skúseností ako s technológiou .NET tak aj s programovací jazykom JAVA som bol priradený do tímu s pracovníkom a mojim konzultantom Matějom Liszkom. Matěj práve využíva pri vývoji webových aplikácií Google Web Toolkit(GWT), ktorý kompiluje JAVA kód na JavaScript.

Neskôr po príchode druhého pracovníka, ktorý s Matějom na projekte pracuje som bol priradený do iného nového projektu ABB MACHSense v tíme s Bronislawom Byrtusom. V tomto projekte sa využíva technológia ASP.NET. Nejednalo sa implementáciu nového projektu ale o refaktORIZÁCIU a doplnenie funkčných požiadavkov do už existujúceho projektu.

3 Použité technológie

Počas odbornej praxe som bol priradený ku dvom rozličným projektom. Prvý projekt využíval framework GWT, ktorý využíva programovací jazyk JAVA.

Druhý projekt využíva technológiu ASP.NET kde bol hlavným programovacím jazykom C# čo bolo pre mňa prívetivejšie keďže som sa s nimi už stretol ako s technológiou tak aj s programovacím jazykom. Pri tomto projekte som bol od začiatku zadania vo firme preto som sa mohol podieľať na každej zmene v tomto projekte nejednalo sa o čisto novú aplikáciu ale zadanie bolo jasné refaktORIZÁCIA a následné zrýchlenie aplikácie neskôr doprogramovanie funkcionality podľa požiadavkou zákazníka.

4 Zadané úlohy ich riešenie

4.1 GWT to HTML Tool

Ide o nástroj pri ktorom som si mal osvojiť GWT keďže som sa s touto technológiou ešte nestretol a tento projekt už bol rozbehnutý dosť dlhý čas. V projekte sa využíva framework UiBinder, ktorý umožňuje robiť presne to čo potrebujeme budovať aplikáciu ako HTML stránky s GWT widgetami. Pomocou tohto frameworku môžeme oddeliť User Interface od kódu JAVA. UiBinder využíva XML pomocou, ktorého sa tvorí UI nachádza sa v ňom HTML tagy, GWT Widgety a ďalšie komponenty.

Tento nástroj mal práve uľahčovať náhľad na vytvorené UI vybranej stránky v UiBinder namiesto kompilovania celého projektu do výslednej webovej aplikácie kde kompilácia zaberala dosť podstatný čas.

Nástroj som vytvoril ako jednoduchú konzolovú aplikáciu, kde sa pri spustení používali dva vstupné parametre jeden pre názov konkrétneho UI, ktoré chceme prekonvertovať do HTML a druhý bol úplná cesta k projektu, z ktorého chceme konvertovať. Výsledkom bolo vytvorená HTML stránka daného UI.

Pri implementácii som musel ako prvé vyriešiť nájdenie konkrétneho súboru kde som pomocou rekurzívneho prehľadávania všetky zložky projektu. Následne načítal nájdený súbor a konkrétne XML a rozparsoval postupne na všetky tagy či už GWT alebo HTML, ktoré obsahovalo. Keďže som s frameworkom GWT dovtedy nemal žiadnu skúsenosť v prvom rade som musel vytvoriť nejakú tabuľku všetkých základných GWT Widgety a Panelov. Následne zistiť na aké HTML tagy konvertuje GWT tieto Widgety a Panely.

Prvý problém nastal pri používaní CSS štýlov keďže v GWT je možné využívať kompilované CSS štýly.

Bolo potrebné tieto štýly načítať z vybraného XML a použiť na každý prekonvertovaný HTML tag výslednej stránky.

Ďalší problém bol využitie vlastných widgety. V UiBinder je možné používať vlastné widgety, ktoré sú uložené v inom XML UiBinder. Tieto widgety museli byť pomocou rekurzívneho načítania tak isto ako hlavná stránka a vložené do výslednej stránky.

Po prekonvertovaní všetkých tagov do HTML bola vytvorená výsledná HTML stránka, ktorej sa prekopírovali použité kompilované štýly a obrázky. Výsledná stránka obsahovala len grafický náhľad danej stránky nie funkčnosť.

4.2 Find unused compiled CSS

V GWT je možné využívať `CSSResource`. `CSSResource` využíva spracovanie CSS v čase kompilácie aplikácie. Odstraňujú nesúlad medzi fyzickými názvami súborov a konštantami v kóde vykonaním kontroly konzistencie počas kompilácie. Použitie `CSSResource` môžeme vidieť na príklade 1.

Úlohou tejto aplikácie je nájsť nepoužité CSS štýly v aplikácii zadané v `Resources`. Ide o jednoduchú konzolovú aplikáciu, ktorej sa pomocou parametra pri spustení odovzdá cesta k projektu kde chceme zistiť nepoužité štýly. Aplikácia pomocou rekurzie nachádza definované `CSSResource` a ďalej hľadá ich využité respektíve nepoužité CSS štýly.

```
public interface ApplicationResources extends ClientBundle {
    public static final ApplicationResources INSTANCE = GWT.create(ApplicationResources.class);

    @Source("Application.css") // Load CSS from Application.css
    Style style();

    public interface Style extends CssResource {

        //Header style
        String floatLeft();
        String floatRight();
        String clearBoth();
        String inline();
        String disabled();

    }
}
```

Výpis 1: Príklad definovania CSS štýlov v `CSSResources`

4.3 Analýza projektu ABB MACHSense

Už po úvodnej analýze projektu sme si všimli, že v projekte sa nevyužíva žiadne rozdelenie vrstiev projekt pozostával len z webu a interných tried. Prístupy do databáze boli vykonávané a spracované väčšinou priamo z code behind webovej aplikácie.

4.4 RefaktORIZÁCIA

Po analýze, ktorá nám zabrala celý týždeň aj s úvodným naštudovaním ako vlastne spustiť projekt sme sa s kolegom Bronislawom Byrtusom pustili do refaktORIZÁCIE. Po dlhej

debate sme sa dohodli, že do solution pridáme ďalšie projekty pre prístup do databázy a užitočné funkcie. Ďalším problémom bolo vybrať akým spôsobom budeme pristupovať do databázy. Po zisťovaní od kolegov sme sa rozhodli použiť PetaPoco. Je to jednoduchá data access vrstva pre .NET aplikácie, ktorá sama vygeneruje C# triedy podľa databázy a dá sa s nimi veľmi jednoducho pracovať pri databázových operáciách ako Update, Select, Delete. Môžeme pracovať s týmito objektmi a nemusíme písať na ne SQL dotazy. Pre PetaPoco sme vytvorili v Solution projekt s názvom Database a ďalší projekt DataAccess, v ktorom boli obsiahnuté metódy, ktoré využívali PetaPoco a zložitejšie SQL dotazy.

4.5 Translators

Pre používanie PetaPoco sa museli vytvoriť prekladače(konvertory), ktoré konvertovali PetaPoco objekty na internal objekty, ktoré obsahovali ďalšie funkcie pre aplikáciu. Vytvorila sa jedna abstraktná trieda BaseTranslator, v ktorej bola hlavná metóda Translate. Z tejto triedy dedili všetky translatory pre objekty, ktoré obsahovali štyri metódy dve metódy na konvertovanie z petapoco objektu na internal objekt a dve metódy na presný opak.

Museli sme vytvoriť translatory všetkých objektov, ktoré sa v aplikácii využívajú čo zabralo okolo 3 dni.

4.6 DataAccess Objekty

Ako som spomenul vyššie chceli sme oddeliť databázové operácie do zvlášť vrstvy ako PetaPoco keďže aplikácia dovtedy používala jednoducho SQL dotazy vo Webovej aplikácii v code behind alebo v objektoch. Tomuto sme sa vyhli pri refaktorizácii a tak sa vytvoril projekt, ktorý obsahoval zvlášť triedy každého objektu, ktorý nejakou spôsobom prístup do databázy. Táto refaktorizácia zabrala ďalší týždeň kde som postupne prepisoval všetky SQL dotazy na databázu do PetaPoco syntaxe a ukladal ich ako metódy.

Príklad prepísaného SQL dotazu do PetaPoco syntaxi, ktoré priamo vracia list PetaPoco objektov.

```
public List<Customer> GetAllCustomers()
{
    Sql query = Sql.Builder.Select("*").From("Customer");
    return _db.Query<Customer>(query).ToList();
}
```

Výpis 2: Získanie zákazníkov z DB pomocou PetaPoco

4.7 Dodatočná refaktorizácia a urýchlenie aplikácie

Po oddelení projektu do databázovej vrstvy som začal pracovať na urýchlení načítania aplikácie po prihlásení užívateľa. Táto úloha mala od zákazníka najvyššiu prioritu. Po analýze som zistil, že stránka po prihlásení používa zbytočné SQL dotazy a následnú prácu s nimi napríklad po prihlásení v aplikácii sa do tabuľky vypisoval počet objektov daného užívateľa. Namiesto jednoduchého SQL dotazu, ktorý by vracal počet objektov jedným číslom pomocou COUNT sa v aplikácii najskôr z databázy načítali všetky objekty do listu objektov a až následne sa zistoval počet objektov v danom liste. Všetky podobné prípady som sa pokúsil odstrániť čo po následnom testovaní poukázalo zrýchlenie aplikácie.

4.8 AddData Collector

Túto a nasledujúcu úlohu som dostal priamo od zákazníka, ktorý na týždeň prišiel navštíviť CZOPC ABB v Ostrave a vysvetliť nám ďalšie nejasnosti ohľadom funkčnosti projektu. Mojou úlohou bolo naimplementovať novú stránku do aplikácie kde by mohol užívateľ jednoducho pridávať a upravovať svoje DataCollector. DataCollector bola jedna tabuľka v databáze čiže splnenie tejto úlohy nebol až taký veľký problém. Jediný problém nastal pri požiadavke triediť DataCollector, ktoré sa zobrazovali v tabuľke podľa stĺpca, na ktorý sa kliklo. Na tento problém som využil knižnicu javascriptu List.js, ktorá už bola v projekte využitá. Keďže som sa s ňou ešte nestretol nejakú dobu mi zabralo naštudovanie potrebnej dokumentácie [4].

4.9 Configure Parameters Css štýly

Ďalšou úlohou od zákazníka bolo doprogramovanie funkcionality a CSS štýlov už do existujúcej stránky. Keďže na stránke sa mali dať upravovať jednotlivé parametre pripojených Machines čo neboli len jednoduché tabuľky v databáze ale reálne stroje mi musel zákazník najprv vysvetliť ako sa dajú tieto parametre pomocou externých ABB knižníc a vytvorených metód v projekte upravovať.

Po vysvetlení som sa rozhodol vytvoriť jednoduchú komponentu, ktorá zobrazovala zoznam užívateľových Machines. A tabuľku, ktorá zobrazuje jednotlivé parametre vybranej Machine, ktorých hodnoty sa dajú upravovať.

4.10 Testovanie aplikácie

Testovanie výsledného programu je jedna s najdôležitejších úkonov, ktoré sa musia vykonať minimálne na konci celého projektu. Keďže firme ABB závisí na kvalite výsledného produktu preto sa museli testy vytvárať a spúšťať všetky aj testy minulých iterácií na konci každej integrácie. Metodika vytváranie testov po každej iterácii sa mi zdala najlepším riešením keď sme mohli odhaliť budúce chybné funkcionality metód po menšom čase ako na konci celého projektu. Keďže aplikácia, ktorú sme dostali do rúk na začiatku nemala dovtedy žiadne testy vytvorené museli sa najprv všetky alebo aspoň podstatná časť vytvoriť.

4.11 MS UNIT Testy

V prvom rade som testoval pomocou Unit Test Framworku od Micorsoftu. Pre tieto testy som vytvoril samostatný projekt v solution. V týchto testoch sa priamo testujú vybrané metódy a triedy. Každá zložitejšia metóda musela byť otestovaná či prinesie rovnaký výsledok aký očakávame. Ak všetky testy prejdú úspešne môžeme povedať o aplikácii, že je bezchybná z logickej časti tým myslím, že metódy robia presne to čo chceme a nemôže dôjsť k chybným výsledkom v aplikácii.

4.12 Coded UI Testy

Ako druhé sa vykonávali CUIT testy. Tieto testy sú druhým odvetvím testovania, ktoré simulujú reálneho užívateľa, ktorý využíva našu webovú aplikáciu. Po dohode s ostatnými kolegami, ktorý pracovali s CUIT testami na svojich projektoch som sa rozhodol nevyužívať automatické vytvorenie testu kedy len naklikám presne čo má test spraviť niečo ako keby nahrám presne načo má kliknúť a už to len opakuje. Použil pomocnú triedu, ktorú vytvorili ďalší kolegovia Viliam a Peťo kde som využil metódy na nájdenie HTML control podľa ID a následne vyplnenie, vybranie alebo kliknutie. Jediný problém nastal pri hľadaní konkrétneho Control podľa ID keďže ASP.net vytvára ID nielen z názvu ale aj ďalších informácií kvôli tomu som musel pri vytváraní testu zisťovať ID pomocou Developer Tool, ktoré obsahuje Internet Explorer alebo pomocou FireBug dodatkového nástroja pre Firefox.

4.13 Technická dokumentácia

Ďalšou dôležitou časťou v každom produkte od firmy ABB je aj technická dokumentácia. Ktorá je dôležitá najmä pre ďalších programátorov, ktorý budú s aplikáciou praco-

vať. Keďže tento projekt, ktorý sme dostali na refaktorizáciu neobsahoval dokumentáciu sme museli vytvoriť túto dokumentáciu pre celý projekt a dohodli sme sa s kolegom Bronislawom, že na každom konci budeme ako aj testy tak aj dokumentáciu dopĺňovať.

Skúsenosti s dokumentáciou som mal už v škole v predmete Základy programování kde sa využíval program Doxygen, ktorý som využil aj v tomto prípade. Doxygen väčšiu časť dokumentácie zvládne sám keďže ju generuje z komentárov kódu. Tie ale trebalo najskôr vytvoriť na tento problém som použil rozšírenie Atomineer Plus, ktorý taktiež vygeneruje všetky komentáre v Solution sám. Predsa je to len program tak trebalo skontrolovať všetky vygenerované komentáre.

Po ďalšej dohode s Bronislawom som k dokumentácii vytvoril inštaláciu príručku pre developerov, ktorá umožní rýchlejšie spustenie projektu pre ďalšiu úpravu. Bez tejto príručky nám trvalo 3 dni kým sme úspešne spustili projekt do funkčnej podoby.

Nakoniec som musel vytvoriť konfiguračný súbor pre Doxygen za pomoci Doxygen Wizard.

5 Záver

5.1 Chýbajúce vedomosti a zručnosti

Moje chýbajúce vedomosti sa prejavili najmä pri práci na prvom projekte kde sa vyžívala technológia GWT, s ktorou som sa dovtedy počas štúdia nestretol. Preto som po príchode do tímu nepracoval priamo na projekte ale na menších nástrojoch, ktoré som vyššie spomínal GWT to HTML a Find Unused CSS, v ktorých som si postupne osvojoval túto technológiu. Ďalej mi chýbala skúsenosť s prácou v tíme čo je väčšinou základom v každej firme nielen ABB. Súčasťou práce v tíme bola aj metóda vývoja SCRUM, o ktorej som počas štúdia len počul. Toto bola len malá prekážka, na ktorú som si po pár dňoch zvykol.

5.2 Zhrnutie bakalárskej praxe

Výber bakalárskej praxe vo firme ABB pred bakalárskou prácou bola podľa môjho pohľadu pre mňa výhodou. Bola pre mňa cennou skúsenosťou kde som si zdokonalil svoje vedomosti a zoznámil sa s prácou vo firme. Som veľmi vdáčný, že som svoju prax mohol vykonávať vo veľkej medzinárodnej firme ABB kde som si mohol zdokonaľovať nielen technické vedomosti ale taktiež aj anglický jazyk, prácu v tíme a riešenie každodenných problémov pri vývoji softwaru.

Tomáš Dižo

6 Literatúra

- [1] Spoločnosť ABB s. r. o. [online]. [cit. 2014-04-11] *Dostupný na*
<http://new.abb.com/about/>
- [2] Google Web Toolkit Documentation [online]. [cit. 2014-04-12] *Dostupný na*
<http://www.gwtproject.org/doc/latest/DevGuide.html>
- [3] .NET Framework Class Library [online]. [cit. 2014-04-12] *Dostupný na*
<http://msdn.microsoft.com/en-us/library/gg145045.aspx>
- [4] List.js dokumentácia [online]. [cit. 2013-04-12] *Dostupný na*
<http://listjs.com/docs>
- [5] Doxygen documentácia [online]. [cit. 2013-04-12] *Dostupný na*
<http://www.stack.nl/~dimitri/doxygen/manual/index.html>